Practical Bluetooth Traffic Sniffing: Systems and Privacy Implications

Wahhab Albazrqaqoe, Jun Huang, Guoliang Xing

Department of Computer Science and Engineering
Michigan State University
Bluetooth Everywhere

- 3 billions Bluetooth devices were shipped in 2015

- Most applications are privacy-sensitive
  - Car telematics
  - Health monitoring
  - Mobile payment
Bluetooth Privacy

- Bluetooth encrypts packets using a 128-bit key
- However, an attacker can ...
  - Sniff the pairing process to retrieve the link key [MobiSys’05]
  - Capture 44.3 MBs of data to crack encryption in $2^{38}$ operations [Crypto’04, Crypto’05]
  - Inspect traffic pattern to track user activity without decrypting packets [CCS’12, HotMobile’16]
Bluetooth Privacy

- Bluetooth encrypts packets using a 128-bit key
- However, an attacker can ...
  - Sniff the pairing process to retrieve the link key [MobiSys’05]
  - Capture 44.3 MBs of data to crack encryption in $2^{38}$ operations [Crypto’04, Crypto’05]
  - Inspect traffic pattern to track user activity without decrypting packets [CCS’12, HotMobile’16]
Is Bluetooth secure to traffic sniffing?
Basic Frequency Hopping

- Switch b/w 79 subchannels randomly
  - 1600 hops/s

- Current hop is defined by
  - Device MAC address (public)
    - A basic hop seq. of $2^{27}$ long
  - A 27-bit logic clock (secret)
  - Index of current hop
Adaptive Frequency Hopping

• Remaps ‘bad’ subchannels
  – MAC address (public)
  – Clock (secret)

• Vendor-dependent behavior
  – No standard definition of ‘good’ and ‘bad’ sub-channel conditions
Sniffing Bluetooth Traffic is Challenging

- Secret clock value
- Don’t know the next hop
Sniffing Bluetooth Traffic is Challenging

- Secret clock value  →  Don’t know the next hop
- Vendor-dependent AFH implementation  →  Difficult to follow adaptive hopping
Sniffing Bluetooth Traffic is Challenging

- Secret clock value
- Vendor-dependent AFH implementation
- Heavy interference in 2.4 GHz band

→ Don’t know the next hop
→ Difficult to follow adaptive hopping
→ Degradation of sniffing performance
State-of-the-Art Bluetooth Sniffers

- Sniff all 79 Bluetooth subchannels in parallel
  - Need to monitor a total of 83.5 MHz spectrum
  - Rely on specialized radios for parallel signal processing
  - Typically cost around $10k to $25k per unit

Nutaq WDXG RF Wideband Digitizer
Icom IC-R9500 wideband receiver
$17,500.00
State-of-the-Art Bluetooth Sniffers

• Sniff all 79 Bluetooth subchannels in parallel
  • Need to monitor a total of 83.5 MHz spectrum
  • Rely on specialized radios for parallel signal processing
  • Typically cost around $10k to $25k per unit

Practical Bluetooth sniffer using general, inexpensive, portable wireless platforms ??
BlueEar: A Dual-Radio Sniffer

- **The Scout radio**
  - Sniffs a single subchannel to acquire clock value
  - Perform basic hoping to learn subchannel status
  - Selectively jam to avoid interference

- **The Snooper radio**
  - Follow adaptive hopping to capture packets
BlueEar: A Dual-Radio Sniffer

• The *Scout* radio
  • Sniffs a single subchannel to acquire clock value
    • Perform basic hoping to learn subchannel status
    • Selectively jam to avoid interference

• The *Snooper* radio
  • Follow adaptive hopping to capture packets
Brute-force Clock Acquisition

- Generate basic hop sequence (BHS)
Brute-force Clock Acquisition

- Generate basic hop sequence (BHS)
- Stay on subchannel 2 to capture the target’s packets
Brute-force Clock Acquisition

- Generate basic hop sequence (BHS)
- Stay on subchannel 2 to capture the target’s packets
- Compare observed pattern w/ BHS at all clock values
Brute-force Clock Acquisition

- Generate basic hop sequence (BHS)
- Stay on subchannel 2 to capture the target’s packets
- Compare observed pattern w/ BHS at all clock values

![Diagram showing basic hop sequence and clock acquisition process.]

- Basic Hop Sequence Generator
- MAC Address
- Subchannel
- Clock
- Match at clock 35
Probabilistic Clock Matching

• **Goal**: Find the correct clock despite remapped packets

Probabilistic Hypothesis Testing

\[
\frac{d_c}{n} - 2\frac{\sigma}{\sqrt{n}} \geq \frac{59}{79}
\]

- **NO**: Remains as a clock candidate
- **YES**: Eliminate incorrect clock

Update mismatch ratio | a packet captured

Mismatch ratio < 59/79
**Probabilistic Clock Matching**

- **Goal**: Find the correct clock despite remapped packets

Update mismatch ratio | a packet captured
-----------------------|-----------------------------
Probabilistic Hypothesis Testing
\[
\frac{d_c}{n} - 2 \frac{\sigma}{\sqrt{n}} \geq \frac{59}{79}.
\]

- **No**: Remains as a clock candidate
- **Yes**: Eliminate incorrect clock

Average latency < 50 sec
BlueEar: A Dual-Radio Sniffer

• The *Scout* radio
  • Sniffs a single subchannel to acquire clock value
  • Perform basic hoping to learn subchannel status
  • Selectively jam to avoid interference

• The *Snooper* radio
  • Follow adaptive hopping to capture packets

---

[Diagram showing the process flow between Scout and Snooper with nodes for Clock Acquisition, Subchannel Classification, Selective Jamming, Basic Hop Selection, and Adaptive Hop Selection.]
Packet-based Channel Classifier

 Sense Channel X

 X clear?

 Yes

 TX on Channel X

 No

 Remap to Channel Y
Packet-based Channel Classifier

1. Scout visits all subchannels
2. Collect packet statistics
3. Compute prob. of ‘bad’ subchannel
Packet-based Channel Classifier

Sense Channel X

X clear?

Yes → TX on Channel X

No → Remap to Channel Y

✓ Robust to vendor-dependent implementations

✗ Need to capture sufficient packets
RSS-based Channel Classifier

- Classify subchannels based on interference condition
- PDF of received signal strength (RSS) as feature

![Graph showing noise floor and interference](image)
RSS-based Channel Classifier

- Classify subchannels based on interference condition
- PDF of received signal strength (RSS) as feature

✓ Low classification delay

✗ Poor accuracy
  - Spectrum context disparity
  - Oblivious to vendor-dependent impl.
Hybrid Classifier

• Leverage accurate packet-based classifier to train SVM
Hybrid Classifier

- Leverage accurate packet-based classifier to train SVM
Hybrid Classifier

- Leverage accurate packet-based classifier to train SVM
Hybrid Classifier

- Leverage accurate packet-based classifier to train SVM
- Employ SVM for responsive classification
- Fallback to packet-based classifier when SVM’s confidence is low
BlueEar: A Dual-Radio System

• The *Scout* radio
  • Sniffs a single subchannel to acquire clock value
  • Perform basic hoping to learn subchannel status
  • **Selectively jam to avoid interference**

• The *Snooper* radio
  • Follow adaptive hopping to capture packets
Selective Jamming

WiFi AP @2433MHz → BlueEar → Bluetooth Target

Bad sniffing on Channel X → Jam X to drive the target away → Enforced cooperation to avoid interference
Implementation

• Implmented on two Ubertooth, costs $160
• Optimized Ubertooth firmware for real-time hopping
• Offloading computation-intensive tasks to commodity laptop

Experiment setup

• 52 Bluetooth devices (headsets, smartphones …)
• Baseline – Basic hopping sniffer, no selective jamming
• Benchmarks with controlled interference
• Collocated with a Wi-Fi based enterprise WLAN
Packet Capture Rate

>90% capture rate w/ interference from a large-scale, commodity Wi-Fi network
Packet Capture Rate

high packet capture rate up to 20 meters
Practical Countermeasure

- Pseudo-randomly flip subchannel status
  - Disturb the clock acquisition algorithm
  - Interfere with subchannel classification
  - Can be implemented using a user-space script
  - No modification to peripheral devices

(a) Without countermeasure
(b) With countermeasure
Conclusion

• **BlueEar: A dual-radio system for Bluetooth traffic sniffing**
  – Sniff Bluetooth traffic based on novel architecture and algorithms
  – Implement using general, inexpensive wireless hardware
  – Maintain high packet capture rate despite complex interference conditions

• **Privacy Implications and Countermeasure**
  – Re-define the bar of Bluetooth traffic sniffing
    • Reducing sniffer cost by **100-fold**
  – Call for research efforts for enhancing Bluetooth security
  – Explored simple and practical countermeasure
Thank you!

Q/A